Against the "networked information economy": rethinking decentralization, community, and free software development

Article (Accepted Version)

AGAINST THE "NETWORKED INFORMATION ECONOMY": RETHINKING DECENTRALIZATION, COMMUNITY, AND FREE SOFTWARE DEVELOPMENT

Ben Roberts

The free, libre, and open source software (FLOSS) movement is often cited as an example of fundamental changes in production associated with the rise of information networks. Don Tapscott and Anthony D. Williams see it as an example of *wikinomics,* a powerful new form of economic production based on sharing and self-organization.[1] Yochai Benkler argues that the form of social or peer production evident in FLOSS can compete with or even displace traditional forms of capitalist organization and sees it as a symptom of the "networked information economy," replacing the "industrial information economy," which has been in force since the late-nineteenth century.[2] The main feature of this networked information economy is the much greater role within it for *decentralized individual action*.[3]

This essay contests this broadly liberal take on the significance of the free software movement. It argues that Benkler is typical of an approach that tends to view the Internet as the primary enabler of decentralization (for example, in the form of decentralized software development). Of course, it would be foolish to discount the role that networks have played in the success of FLOSS. But, as I will argue, an examination of its history suggests that its origins can't be entirely accounted for in purely technological and economic grounds. The proponents of such an approach often resist the claim that their focus on networks is technologically determinist,[4] arguing that the network itself is shaped by social and cultural factors. However, this makes the explanatory power of networks rather circular: the existence of networks explains social and cultural behavior at the same time as being explained by it. The explanation of free software in terms of the economic properties of networks is simply too static and asocial; it serves to explain how a particular mode of production can functionally compete with the market without really being able to account for how this change has come about. In particular, as I will show, the marginalization

Criticism Summer 2011, Vol. 53, No. 3, pp. 385–405. ISSN 0011-1589.

of the question of labor in theories of peer production disconnects the analysis from wider changes in the nature of economic production. Moreover, the emphasis on the empowerment of individuals ignores the possibility that decentralization consists in radical new forms of sociality and collectivity, something that can perhaps be seen in the communities built around particular free software projects. As I will argue, Gilbert Simondon's concept of *individuation* may help inform a better understanding of the emergence of such free software communities.

From the perspective of a nonprogrammer, it is easiest to understand the concept of FLOSS by explaining how proprietary software such as Microsoft Windows is distributed. The operating system and applications that users run on their personal computers (PCs) are most often *compiled* software. The creators of the application have written the code in any one of a number of different human-readable programming languages; this program is known as the *source code*. The source code is then converted by another program known as a compiler into *object* or *binary code,* which is a machine-readable version of the program executed when the application is run on a PC. Binary, or object, code is the form in which software is most often distributed to users, either on installation media such as compact discs (CDs), installed on a computing device before sale or, increasingly, downloaded over the Net. The end user of proprietary software created and distributed in this fashion has no access to the original source code and is therefore unable to alter, change, or fix the software running on their machine: in the words of Bob Young, the former chief executive officer (CEO) of open source software company Red Hat, purchasing proprietary software is like buying a "car with the hood welded shut."[5]

In addition to the absence of source code, proprietary software is usually distributed under the terms of a highly restrictive software license. In effect, the applications run by a user remain the property of the software maker and may be used only under the terms of this contract. Typically, the license prevents sharing or redistribution of the software and any form of tampering or *reverse engineering:* that is, taking the product apart in order to discover how it has been made. Companies such as Microsoft therefore not only own the means of production of software but also exert a tight and perhaps unprecedented degree of control over the distribution and consumption of their products.

Software created and distributed under a FLOSS license, on the other hand, guarantees users access to the source code and therefore the ability to modify the programs they use. In addition, the license grants users freedom to copy and redistribute the software. So-called *copyleft* licenses, such as the GNU (GNU's Not Unix) Public License (GPL),[6] make a further

stipulation requiring that modifications to the software, if redistributed, must also be made available under the same license. In effect, they require programmers modifying the original code to make their changes available to the community. As Steven Weber puts it, "[T]he GPL uses copyright law to ensure that free software *and derivative works from free software* remain free."[7]

These licenses legislate two radically different approaches to the production of software. On the one hand, proprietary software is usually manufactured by a single company that derives an income—and pay programmers—from the proceeds of license sales. On the other hand, in the FLOSS approach, software is created by a community consisting of a mixture of unwaged volunteers and paid programmers; these latter employees often work for multiple (and competing) companies, and their wages are usually paid for through the proceeds of a related product (support or some other kind of service) rather than through license sales. Proprietary software is a paid-for product, manufactured by private companies, that is made and marketed like any other commodity. FLOSS is distributed for free and often produced by a disparate, distributed collection of developers.

Given the dominance of what Mark Fisher has called "capitalist realism," one might find it surprising that a collective form of production with property held in common—one that the former Microsoft CEO Bill Gates has described as "communism"—could compete with the more recognizable form of capitalist production represented by the proprietary model.[8] Yet free software has been extraordinarily successful. In April 2010, a survey by the network monitoring firm Netcraft estimated that the open source Web server Apache was being used by over half of all active websites on the network.[9] The operating system Linux is used to operate the networks of some of the biggest companies on the Web, including Google and Amazon. Yet perhaps even more surprising is the sheer scale of the software currently emerging from FLOSS projects. Software lines of code (SLOC) is used to gauge the size of software projects based on the number of lines of source code written by their programmers. SLOC can be used to estimate the amount of labor time that has gone into a particular application. Debian GNU/Linux is a community-produced distribution of the Linux operating system. (*Distribution* here means that Debian consists not only of the core operating system [or kernel] but also of the wide range of applications that can be run on top of the Linux system [e.g. Web browsers, office software, and server applications], which can be freely installed from Debian's large software repository.) Researchers at King Juan Carlos University in Madrid have used automated line-counting

tools to generate line counts for programs in Debian's software repository. These figures give some indication of how much labor is currently being expended in FLOSS endeavors (table 1).

From this table we can see that the amount of software in Debian's repositories has grown by a factor of over 10 in the last ten years and that increase in code is equivalent to about 110 thousand person-years of combined programmer effort; had all that labor been paid for by companies, it would have cost, on these estimates, over $300 billion. Clearly, both the success and the productivity of open source poses a number of puzzles for the perspective of "capitalist realism." How is collective and collaborative labor of the kind that goes on in FLOSS able to compete with, and in many cases prevail over, the competitive rigor of labor organized by the market? How is it that competing software companies now find themselves sharing code contributions with one another? What are we to make of the profitable businesses that have been built on top of largely unpaid labor?

Over the past ten years, a response has emerged in the work of writers such as Yochai Benkler, Don Tapscott, and Anthony D. Williams. These writers see the success of FLOSS as essentially bound to the arrival of low-cost distributed communications networks. They downplay the social and political aspects of free software in favor of an analysis of the functional and business advantages of shared intellectual property and mass collaboration (or what Benkler calls peer production). Their approach is therefore in stark contrast with writers, such as David M. Berry, who see in free software new forms of political imaginary or, as he puts it, the promise of "a form of collective decision-making through a democratization of technology."[11]

In effect this argument echoes one that is at the heart of the FLOSS movement between the proponents of *free software* and those of *open source*. This dispute is often encapsulated in terms of the respective positions of two important figures within the evolution of FLOSS: Richard Stallman and Eric Raymond. Stallman, founder of the Free Software Foundation, is often seen as espousing an *ethical* approach to what he prefers to call free software. As Chris DiBona, Sam Ockman, and Mark Stone put it,

> Stallman launched the GNU project because essentially he feels that the knowledge that constitutes a running program—what the computer industry calls the source code—should be free. If it were not, Stallman reasons, a very few, very powerful people would dominate computing. Where

Table 1. SLOC counts, estimated labor time, and cost for the Debian GNU/Linux distribution[10]

Version	Number of software packages included	Software lines of code	Estimated combined labor time (person-years)	Estimated equivalent development cost
Debian 2.0 (1998)	1,096	25267766	8,389.29	$1.13 billion
Debian 3.0 (2005)	8,560	215812764	79,764.74	$175.3 billion
Debian 5.0 (2009)	12,002	323551126	122,030.97	$315.2 billion

> proprietary commercial software vendors saw an industry guarding trade secrets that must be tightly protected, Stallman saw scientific knowledge that must be shared and distributed.[12]

Raymond, on the other hand, emphasizes the practical benefits of sharing source code. Along with Bruce Perens, he founded the Open Source Initiative (OSI) in 1998, which promotes a more business-friendly approach to FLOSS. In his seminal essay "The Cathedral and the Bazaar," Raymond contrasts the centralized *cathedral* approach to software development usually found until then in the software industry with what he terms the distributed *bazaar* style of programming, which he believes to characterize FLOSS programming and, in particular, the evolution of the Linux operating system.[13] Raymond relates his early experience of Linux as follows:

> Linux overturned much of what I thought I knew. . . . [T]he Linux community seemed to resemble a great babbling bazaar of differing agendas and approaches . . . out of which a coherent and stable system could seemingly emerge only by a succession of miracles. The fact that this bazaar style seemed to work, and work well, came as a distinct shock. . . . [T]he Linux world not only didn't fly apart in confusion but seemed to go from strength to strength at a speed barely imaginable to cathedral-builders [i.e., traditional software developers].[14]

As is clear from Raymond's opening remarks, "The Cathedral and the Bazaar" describes open source as a software development practice rather than extolling the beliefs and commitment to freedom so important to the Free Software Foundation. Instead of posing the question, "Why should software development be free?" it asks, "Why does open source work? What make it successful?"[15] Raymond's essay has been highly influential both inside and outside the software world. It is widely credited, for example, with influencing Netscape's decision to open the source of its Web browser, code that eventually became the Firefox Web browser. But equally his essentially pragmatic approach to FLOSS advocacy seems to have influenced academic discussion of the phenomenon in legal and communication studies, an influence that can be seen at work in the arguments of scholars such as Lawrence Lessig and, as I will show here, Yochai Benkler.

Benkler's approach can perhaps be best seen in his 2002 article "Coase's Penguin, or, Linux and *The Nature of the Firm*."[16] Benkler's aim from the outset is to analyze free software in terms of the wider phenomenon that he refers to as "commons-based peer production." Therefore he opens by acknowledging that he is not interested in the cultural specificity of free software:

> Rather than trying to explain what is special about software or hackers, I generalize from the phenomenon of free software to suggest characteristics that make large-scale collaborations in many information production fields sustainable and productive in the digitally networked environment without reliance either on markets or managerial hierarchy.[17]

Benkler's move here is typical in that it seeks to understand free software in terms of a wider phenomenon associated with distributed communications networks. Indeed, most of the examples in his article are drawn from outside the software world: *Wikipedia,* NASA Clickworkers, and Web communities such as Slashdot and Kuro5hin.[18] The common ground, or perhaps commons ground, in these examples is that in each case communities are engaged in a collaborative production of knowledge. In essence Benkler seeks to outline an economic rationale for this phenomenon. His account draws in no small part on Ronald Coase's arguments about organizational structure in the well-known article "The Nature of the Firm."[19] Coase's is a classic explanation for the existence of firms, companies, or corporations in market economies. We are, perhaps, so used to associating private companies and corporations with capitalist economies that it might seem rather counterintuitive to see their coexistence as problematic. Yet although companies can be seen as competing in free markets, within their own organization they are classically organized in the hierarchical control structures associated with Soviet-style command economies. The puzzle, then, is why, if the pricing model provided by markets is so efficient, do individuals come together in the form of companies rather than simply trading their labor on the free market? Or as Coase puts it, the question is "why organizations such as firms exist in a specialized exchange economy in which it is generally assumed that the distribution of resources is 'organized' by the price mechanism."[20] The answer that Coase provides is that there is an overhead to operating on the open market, *transaction costs* associated with the discovery of prices and the implementation of contracts. This means that on some occasions it

may be cheaper to organize a given transaction with the firm rather than on the open market: "[T]he operation of a market costs something and[,] by forming an organization and allowing some authority (an 'entrepreneur') to direct the resources, certain marketing costs are saved."[21] So, essentially, capitalist economies are characterized for Coase by two methods of organization: firm-organized production and market-organized production.

The crucial insight of applicability to free software here is that firms and markets are not always the cheapest ways of organizing production. Benkler argues that we should treat both firms and markets as "mechanisms by which individual agents reduce uncertainty as to the likely value of various courses of productive action."[22] This leads him to hypothesize that under certain circumstances a third mode of organization, *peer production,* may be able to supplant both the market and the firm. As Benkler puts it, "[W]hen the cost of organizing an activity on a peered basis is lower than the cost of using the market or hierarchical organization, then peer production will emerge."[23] The conditions of this emergence are essentially fourfold and tied to the emergence of what he calls the networked information economy: Firstly, information as an object of production is nonrivalous, meaning that its availability to one person does not impact its availability to another (unlike, say, a house or a car). Secondly, cheap computing power means that capital costs of production are low. Thirdly, the creative labor involved is highly differentiated and the individuals involved have a more sophisticated understanding of what they can contribute to a given project than either a hierarchical organization or market could take account of or master. Fourthly, low communications costs allow much easier coordination of productive efforts. Benkler is hardly alone in arguing that the Internet has lowered transaction costs, making it easier to organized production outside the firm. Indeed, we can find this view enthusiastically endorsed by Tapscott and Williams:

> [T]he Internet has caused transaction costs to plunge so steeply that it has become much more useful to read Coase's law, in effect, backward: Nowadays firms should shrink until the cost of performing a transaction internally no longer exceeds the cost of performing it externally.[24]

There are a number of weaknesses in this argument as an account of the free software movement, questions that I address later in this essay. However, I want to focus first on the dominant aspect of this approach, which ties the emergence of new forms of organizing production to the

availability of low-cost network communications. This is the argument that Benkler develops in *The Wealth of Networks, which* argues for the emergence of a new form of information economy, a networked information economy replacing the industrial information economy that has been in force since the late-nineteenth century.[25] The primary feature of this networked information economy is the much greater role within it for decentralized individual action. This empowerment of individuals is the result of two key changes in the new networked information economy. The first is a dramatic reduction in or, as Benkler has it, the "practical elimination of," communication costs. The second is a change in the topology of information networks from the hub-and-spoke model of mass media to a "distributed architecture" with "multidirectional connections."[26] Together these changes enable a new, more democratic and participative, form of political communication that Benkler calls the "networked public sphere." In his depiction of these changes, Benkler is, of course, not alone—arguments about their potential to transform the public sphere have always formed an important part of the way information networks are discussed.[27] Indeed, Benkler's work illustrates two tropes consistently associated with writing about the Internet: The first is the understanding of the network as essentially distributed and decentralized. The second is the assertion that this decentralization leads to a freer form of communication, liberated from state and other controls. In fact it is really corporate control that most concerns Benkler. What open source offers is a form of production that maximizes individual freedom in a way that markets and firms are unable to. Such a claim is echoed by Tapscott and Williams:

> Though it is unlikely that hierarchies will disappear in the foreseeable future, a new form of horizontal organization is emerging that rivals the hierarchical firm in its capacity to create information-based products and services, and in some cases, physical things. . . . [T]his new form of organization is known as peering.[28]

Although this approach essentially embraces political and economic liberalism, there are alternative, more radical visions of decentralization. Theoretical approaches to networks have long laid claim to a concept of the rhizome drawn from Gilles Deleuze and Félix Guattari's *Mille Plateaux*. Here the arborescent model of knowledge, vertical and hierarchical, is contrasted with the horizontal rhizome: "[U]nlike trees or their roots, the rhizome connects any point to any other point," comment Deleuze and Guattari.[29] Computer networks and the Web in particular have seemed

to many to be an incarnation of the rhizome. Kathleen Burnett argued in 1993, "Telecommunications systems are rhizomorphic, as are computer networks. Think of maps you have seen and descriptions you have heard of the internet as rhizome. If we accept the rhizome as a metaphor for electronically mediated exchange, then hypertext is its apparent fulfillment."[30] However, there are several difficulties with the appropriation of the rhizome to describe computer networks. One problem is that the Internet, and the hypertextual Web built on top of it, are perhaps not as rhizomatic as Burnett envisaged. Network research by Albert-László Barabási and others shows that the evolution of the network demonstrates the emergence of concentrated clusters and hierarchies of nodes within apparently flat networks.[31] Does the emergence of hierarchies in this manner then undermine the rhizomatic view of the network? On the one hand, perhaps no. As Deleuze and Guattari themselves point out in *Mille Plateaux,*

> [T]here are knots of arborescence in rhizomes, and rhizomatic offshoots in roots. Moreover, there are despotic formations of immanence and channelization specific to rhizomes, just as there are anarchic deformations in the transcendent system of trees, aerial roots and subterranean stems. The important point is that the root-tree and canal-rhizome are not two opposed models: the first operates as a transcendent model and tracing, even if it engenders its own escapes; the second operates as an immanent process that overturns the model and outlines a map, even if it constitutes its own hierarchies, even if it gives rise to a despotic channel.[32]

However, on the other hand, analyses of the emergent hierarchies within distributed networks may problematize some of the liberatory rhetoric associated with the network as rhizome. The development of hierarchies within online discussion may make us suspicious of the stronger claims made about the distributed nature of the Internet leading to an inherently more democratic and participatory culture. As Tony Sampson argues, it may be better to see the topology of the network as accidental, a mixture of controlled and uncontrolled events that "limit the free action of the network, and subsequently reduce the explanatory power of a distinct model (including the rhizome) to describe the network experience as a whole."[33]

A slightly different approach to this question emerges from Alexander Galloway and Eugene Thacker in *The Exploit,* who argue that "the

liberation rhetoric of distributed networks . . . is a foil for the real workings of power today."[34] They see the distributed, decentralized, or rhizomatic aspects of the information network not as inherently disorganized or apolitical but as amenable to various systems of control: "[R]hizomatics and distribution signal a new management style, a new physics of organization that is as real as pyramidal hierarchy, corporate bureaucracy . . . or any other principle of social and political control."[35] For Galloway and Thacker it is important to understand how control functions within distributed networks, whether electronic or biological, the "quandary" that "no one controls networks, but networks are controlled."[36] Such an understanding requires, on their reading, a new "political ontology of networks." Their approach marks a break not only with more liberatory approaches to distributed networks but also with some aspects of network science.

In particular, Galloway and Thacker take their distance from *graph theory,* which models networks in terms of a set of *nodes* or vertices and a set of *edges* or lines that connect pairs of vertices. They cite approvingly Gilbert Simondon's argument that we should "understand the individual from the perspective of the process of individuation rather than the process of individuation by means of the individual."[37] To put this back in terms of graph theory, the important thing is to understand the network as a process where nodes and edges are individuated rather than understanding the network as essentially being composed of nodes and edges. Galloway and Thacker trace the concept of individuation to Aristotle and gloss it as "what makes a thing what it is."[38] However, it is largely Gilbert Simondon's rethinking of individuation that they have in mind:

> In the context of networks, individuation will have to be understood differently. Instead of the classical definition, in which individuation is always concerned with the production of individuals (be they people, political parties, or institutions), in the control society, individuation is always concerned with the individuation of networks as a whole and the individuation of the component parts of the network.[39]

The consequence is that, instead of deriving political implications from the topology of networks, we see the topology of networks as derived essentially from a process of individuation that is implicitly political.

Such an account demonstrates the limitations of Benkler's approach to understanding peer production in terms of the properties of distributed

communications networks. By overstating the benefits of decentralization, it misses the more complex topological account that Galloway and Thacker outline. A further problem with Benkler's model is that it effectively reduces the political, social, and cultural aspects of peer production by focusing on the economic and functional benefits. This is particularly apparent in the case of the free software movement, which, as I have indicated earlier, has always had a political and ethical dimension quite separate from the pragmatic and processual benefits Benkler prefers to focus on. Indeed, the creation of the GNU project (1984) significantly precedes the widespread adoption of Internet networking, and its founder, Richard Stallman, was motivated partly by a desire to defend a kind of software-engineering practice that was common in the 1960s and 1970s where software source code would be freely circulated by computer scientists on magnetic tape, modifications made, and passed on. This early form of peer production therefore predated both the decline in capital costs associated with the PC and the fall in communication costs associated with distributed networks.

A clue to this problem can be found within Benkler's account of the economics of peer production considered earlier in this essay. To recall, Benkler cites one of the four conditions of the emergence of peer production found in the particular type of labor involved in information production:

> [T]he primary human input—creative talent—is highly variable, more than traditional labor and certainly more than many material resources usually central to production. Moreover, the individuals who are the "input" possess better information than anyone else about the suitability of their talents and their level of motivation and focus at given moment to given production tasks.[40]

Benkler points to this condition to highlight peer production's provision of better information about "human capital."[41] (This fits with his general argument that the firm, the market, and peer production are really modes of information as much as modes of production.) However, the existence of this new form of *creative* labor cannot be reckoned for entirely through the organizational advantages of peer production. To understand this new mode of production, we might need to consider a fundamental shift in the nature of labor, not just a change in the way it is organized. Benkler's neglect of labor underlines the fundamental weaknesses of his argument. He sees the development of the network resulting in new forms

of nonmarket peer production. But the emergence of creative labor may be seen as both preceding the "network information economy" and as representing a transformation *within* market-based capitalist production rather than a step beyond it. As Tiziana Terranova puts it,

> [T]he conditions that make free labor an important element of the digital economy are based in a difficult, experimental compromise between the historically rooted cultural and affective desire for creative production (of the kind more commonly associated with Gilroy's emphasis on "individual self-fashioning and communal liberation") and the current capitalist emphasis on knowledge as the main source of value-added.[42]

As Terranova goes on to argue, such labor can be understood as a fundamental dimension of postindustrial societies, a form of work that Maurizio Lazzarato describes as "immaterial labor."[43] *Immaterial labor,* defined as "the labor that produces the informational and cultural content of the commodity,"[44] is associated by Lazzarato with a transformation that begins at the start of the 1970s and involves "a transformation of working-class labor into a labor of control, of handling information, into a decision-making capacity that involves the investment of subjectivity." Such labor may involve "a series of activities not normally recognized as "work," such as "defining and fixing cultural and artistic standards, fashions, tastes, consumer norms, and, more strategically, public opinion."[45] Immaterial labor is no longer defined by the space of the factory:

> The location in which it operates is outside in the society at large, at a territorial level that we could call "the basin of immaterial labor." Small and sometimes very small "productive units" (often consisting of only one individual) are organized for specific ad hoc projects, and may exist only for the duration of those particular jobs. The cycle of production comes into operation only when it is required by the capitalist; once the job has been done, the cycle dissolves back into the networks and flows that make possible the reproduction and enrichment of its productive capacities.[46]

Lazzarato's account of immaterial labor, then, shares with Benkler the assumption of a break with the firm as a mode of organization. However, it differs in two distinct fashions. Firstly, as has been suggested, immaterial

labor is understood as a change within capitalist market economies rather than as the break with the market form or organization, which Benkler ascribes to peer production. As the many successful companies that have built their business around open source software (such as Red Hat and Google) demonstrate, new forms of labor can still be exploited by market-organized economies. As David Berry puts it, "[I]t is of crucial importance that the FLOSS practices are understood to be taking place within an intensification of capitalism rather than signaling its replacement."[47] The second difference is Lazzarato's insistence that "immaterial labor constitutes itself in forms *that are immediately* collective," or, as Terranova puts it, "[k]nowledge labor is inherently *collective,* it is always the result of a collective and social production of knowledge."[48] Benkler's approach, on the other hand, insists on seeing the phenomenon of peer production as essentially the self-organization of *individuals.* In many ways, this is a direct result of the liberal orientation of his work, one that is made explicit in *The Wealth of Networks:*

> I am offering a liberal political theory, but taking a path that has usually been resisted in that literature—considering economic structure and the limits of the market and its supporting institutions from the perspective of freedom, rather than accepting the market as it is, and defending or criticizing adjustments through the lens of distributive justice. . . . [M]y approach heavily emphasizes individual action in nonmarket relations. Much of the discussion revolves around the choice between markets and nonmarket social behavior. . . . [W]hat is special about our moment is the rising efficacy of individuals and loose, nonmarket affiliations as agents of political economy.[49]

Indeed, this methodological statement summarizes Benkler's approach to understanding phenomena such as peer production or free software in terms of the *nonmarket* exercise of *individual* freedom. However, I would like to suggest that it may be more productive to see free software as the manifestation of neither the "rising efficacy of individuals and loose, nonmarket affiliations" nor a new capitalist exploitation of collective immaterial labor. Rather, we need to understand the free software movement, as Galloway and Thacker suggest in relation to networks, as a new form of individuation.

Indeed, I suspect that Benkler is guilty here of, in Simondon's terms, "thinking individuation from the perspective of the individual." What Benkler misses in his analysis is that in free software production the place

of the firm does not simply disappear but rather gets replaced by the *project,* the particular piece of software or group of software applications that is being worked on. The project may be simply defined by the code itself and a loose network of contributors and users of that software. Alternatively it may be formally codified and linked to more traditional forms of nonprofit organization (such as is the case with the GNOME [GNU Network Object Model Environment], Mozilla, and Apache Software foundations).[50] But in each case the project is substantially more than simply a loose affiliation of individuals, one that usually goes beyond simply the organization of production and includes aspects of social and cultural identification. One recent contribution to thinking about these types of bond is Christopher Kelty's notion of the "recursive public."[51] Kelty uses *recursive* to indicate, among other things, that "this kind of public includes the activities of making, maintaining, and modifying software and networks, as well as the more conventional discourse that is thereby enabled."[52] As Kelty argues,

> [G]eeks use technology as a kind of argument, for a specific kind of order: they argue about technology, but they also argue through it. They express ideas, but they also express infrastructures through which ideas can be expressed (and circulated) in new ways.[53]

Indeed, Kelty's arguments around "recursive publics" set out a much richer set of relationships among free software developers than the narrow economic analysis of peer production is capable of disclosing. Kelty's work suggests the need for a greater focus on the rich and varied nature of "hacker culture" and an understanding of free software projects as cultural institutions.[54]

As Steven Weber argues, the problem with many arguments about the effects of reduced communication and transaction costs is that they focus only on the destructive changes to traditional business organization, ignoring the new structures that have evolved to replace them:

> "[S]elf-organization" as a concept by itself can't fill this gap. The Internet does not solve foundational political problems of organizing complexity. It does not create working divisions of labor. Reducing or even removing the costs of geographically widespread and time-dependent collaboration is important, but that effort still leaves other collaboration costs unsettled—decision making, human emotion, resolution of technical uncertainties, and so on.[55]

Organization structures in FLOSS, Weber suggests, go beyond traditional models of governance and can be seen at work even in the way the code is structured (for example, in the modular design of the Linux kernel) and in the licenses themselves as explicit social structures.

Work such as Kelty's and Weber's suggests that free software production needs to be rethought not as a liberation of individual knowledge creation from the restrictions of the market and the firm but rather as the emergence, or individuation, of new forms of collectivity. Such collectivities can be seen as a liberation from the constraints of existing forms of social organization but also as the source of new forms of power and control. (This parallels the approach taken by Galloway and Thacker in their arguments around protocol as "a materialized functioning of distributed control".[56]) For that reason alone, we can't oppose decentralized individualism with an enthusiastic and naïve embrace of collectivism and collaboration. Indeed, such a position is arguably the other kind of liberation rhetoric associated with much writing about the Internet.[57]

What is needed is to understand the individuation of new forms of collaborative organization free from simplistic teleological accounts of the decentralized individual or network collectivism. Individuation is the key concept here, requiring us to think through further the politics of individuation in relation to Simondon's work. As Alberto Toscano has argued, what is crucial here is the status of the preindividual, that which precedes individuation.[58] Toscano finds three types of reading at work in responses to Simondon. The first interprets the *preindividual* as an "unresolved charge, carried by the individual as a potential, linking this concept to those of human nature and living labor." Toscano links this to the work of Paolo Virno.[59] The second sees it as a preexisting commonality between the individual and collective, which Muriel Combes calls the "intimacy of the common."[60] The third reading that Toscano explores at length is the Deleuzian one. Here the preindividual, as that which precedes individuation, is neither a prior collectivity nor a potential but rather a disparate set of singularities. As Toscano puts it,

> Simondon and Deleuze offer a conception of politics as the invention of a communication between initially incompossible series; as invention of a common that is not given in advance and which emerges on an ontological background of inequality. . . . We are not dealing with the expression of a potential but with the invention of a communication responding to the larval emergence of divergent energies in the social field. The element of politics as analysis and

> intervention is not the genesis and the concretization of social relations, but metastability (or disparation) "as such" and the event-invention which crystallizes it into a new configuration (carrying a further preindividual charge).[61]

What precedes individuation is therefore not something essentially human or a kind of primordial collectivity but rather an unthinkable disparation or inequality.[62] The preindividual charge understood in this third way is not so much a latent force within the social but rather affords insight into the essential instability of social relations. Toscano goes on to give the example from Simondon's *L'individuation psychique et collective* where revolutionary change is figured as a breaking of metastability, a deindividuation or disparation at the heart of the metastable group, which then leads to the crystallization of a new set of social relations. (Simondon suggests this crystallization can be precipitated by the germ or crystal of the revolutionary idea.)[63]

Toscano's interpretation of Simondon enables us to understand the free software movement as emerging *neither* from a network of decentralized individuals *nor* as the expression of a latent force of collective labor. The free software movement is a symptom of the deindividuation of existing types of social relation and relations of production rather than a liberation from them, for a liberation invariably implies some force prior to these institutions (e.g., individual freedom or collective intelligence) that is in the process of being unharnessed. Instead, this is a process in which new types of individual and collective individuation emerge from the deindividuation of older forms. The most important conclusion that one can draw from the arguments about transaction costs is not the economic viability of peer production as decentralized individual action but rather that "the firm" was never *firm* but only ever a "metastable" entity. The cultural politics of free software need to be understood as the *process* in which this entity dissolves and others crystallize, rather than from the perspective of the individual or the collective as a priori.

Such a conclusion underlines the importance of understanding capitalist production as an ever-changing and historically specific formation rather than a teleological ideal. This is why the neglect of the question of labor in Benkler's account is so telling and so significant. Benkler understands that peer production is possible only because of a change in the nature of the product being produced and the type of "creative" labor required, but places this whole question at the margins of his account, preferring to emphasize instead transaction costs and intellectual property regimes. He is not alone in this neglect of labor. As Andrew Ross points

out, it is one of the hallmarks of much recent critical work on questions of intellectual property and commons production.[64] But labor is the crucial pivot that links the questions raised by free and open source software production to wider social and cultural formations, rather than a narrow technological idealism of "the network." The emergence of free and open source software is in many ways a surprising, unpredictable development and not an inevitable outcome of technological change or decentralization. In understanding this development, we need to place less emphasis on the forces of technological or economic necessity and more on its contingent foundations.

Ben Roberts is lecturer in media studies at the University of Bradford. He is currently editing a special issue of the journal New Formations *on the work of Bernard Stiegler.*

NOTES

1. Don Tapscott and Anthony D. Williams, *Wikinomics: How Mass Collaboration Changes Everything* (New York: Penguin, 2006), 25.
2. Yochai Benkler, *The Wealth of Networks: How Social Production Transforms Markets and Freedom* (New Haven, CT: Yale University Press, 2006), 3.
3. Ibid.
4. Ibid., 16–18, 369, 372.
5. Kelly Woo, "Red Hot Red Hat," *Chronicle,* 27 October 1999, http://dukechronicle.com/article/red-hot-red-hat.
6. According to Lawrence Rosen,

   > [T]he authors of the GPL [GNU General Public License] coined the term *copyleft* to describe its license bargain. It is both a play on the word *copyright* and an acknowledgement that it promoted a radical (i.e., *left-wing,* perhaps) departure from traditional licensing models. The role of a *copyleft* software license is to grow the public commons of software rather than allow each owner's *copyright* to pull from the commons. (*Open Source Licensing: Software Freedom and Intellectual Property Law* [Upper Saddle River, NJ: Prentice Hall, 2005], 105; see also Free Software Foundation, "The GNU General Public License," version 3, 29 June 2007, www.gnu.org/licenses/gpl.html)

7. Steven Weber, *The Success of Open Source* (Cambridge, MA: Harvard University Press, 2005), 48.
8. Mark Fisher, *Capitalist Realism: Is There No Alternative?* Zero Books series (Ropley, UK: O Books, 2009); and Michael Kanellos, "Gates Taking a Seat in Your Den," *CNET News,* 5 January 2005, http://news.cnet.com/Gates-taking-a-seat-in-your-den—page-4/2100-1041_3-5514121-4.html?tag=mncol.
9. Netcraft, "April 2010 Web Server Survey," April 2010, http://news.netcraft.com/archives/2010/04/15/april_2010_web_server_survey.html.

10. Grupo de Sistemas y Comunicaciones (GSyC) [System and Communication Group], "Debian Counting," n.d., http://libresoft.dat.escet.urjc.es/debian-counting/.
11. David Berry, *Copy, Rip, Burn: The Politics of Copyleft and Open Source* (London: Pluto Press, 2008), 198.
12. Chris DiBona, Sam Ockman, and Mark Stone, introduction to *Open Sources: Voices from the Open Source Revolution,* ed. Chris DiBona, Sam Ockman, and Mark Stone (Sebastopol, CA: O'Reilly, 1999), 2.
13. Eric S. Raymond, *The Cathedral and the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary,* rev. and expanded ed. (Sebastopol, CA: O'Reilly Media, 2001).
14. Ibid., 21–22.
15. I am simplifying the differences between Stallman and Raymond here. For a more detailed, discourse analysis–led approach to understanding their respective positions, see David M. Berry's excellent account in *Copy, Rip, Burn* (147–87).
16. Yochai Benkler, "Coase's Penguin, or, Linux and 'The Nature of the Firm,'" *Yale Law Journal* 112, no. 3 (2002): 369–446.
17. Ibid., 374.
18. Slashdot.org, "Slashdot: News for Nerds, Stuff That Matters," n.d., http://slashdot.org/; and kuro5hin.org, "Kuro5hin: Technology and Culture, from the Trenches," n.d., www.kuro5hin.org/.
19. R[onald] H[arry] Coase, "The Nature of the Firm," *Economica* 4, no. 16, n.s. (1937): 386–405.
20. Ibid., 393.
21. Ibid., 392.
22. Benkler, "Coase's Penguin," 406.
23. Ibid., 403.
24. Tapscott and Williams, *Wikinomics,* 56.
25. Benkler, *Wealth of Networks,* 3.
26. Ibid., 212.
27. See Peter Dahlgren, "The Internet, Public Spheres, and Political Communication: Dispersion and Deliberation," *Political Communication* 22, no. 2 (2005): 147–62.
28. Tapscott and Williams, *Wikinomics,* 23.
29. Gilles Deleuze and Félix Guattari, *A Thousand Plateaus: Capitalism and Schizophrenia,* trans. Brian Massumi (Minneapolis: University of Minnesota Press, 1987), 21.
30. Kathleen Burnett, "Toward a Theory of Hypertextual Design," *Postmodern Culture* 3, no. 2 (1993): 1–28, quotation on 16, http://muse.jhu.edu/journals/postmodern_culture/v003/3.2burnett.html.
31. See Albert-László Barabási and Réka Albert, "Emergence of Scaling in Random Networks," *Science* 286, no. 5439 (1999): 509–12; Andrei Broder, Ravi Kumar, Farzin Maghoul, Prabhakar Raghavan, Sridhar Rajagopalan, Raymie Stata, Andrew Tomkins, and Janet Wiener, "Graph Structure in the Web," *Computer Networks* 33, no. 1 (2000): 309–20; and Lada A. Adamic and Bernardo A. Huberman, "Power-Law Distribution of the World Wide Web" [Abstract], *Science* 287, no. 5461 (2000): 2115a.

32. Deleuze and Guattari, *A Thousand Plateaus,* 20. The remarks here might point to what I think Charles J. Stivale sees as a more cautious understanding of the rhizome in Deleuze and Guattari (*The Two-fold Thought of Deleuze and Guattari: Intersections and Animations* [London: Routledge, 1998], 83–89).
33. Tony Sampson, "The Accidental Topology of Digital Culture: How the Network Becomes Viral," *Transformations,* no. 14 (2007), www.transformationsjournal.org/journal/issue_14/article_05.shtml.
34. Alexander R. Galloway and Eugene Thacker, *The Exploit: A Theory of Networks,* Electronic Mediations series (Minneapolis: University of Minnesota Press, 2007), 16.
35. Ibid., 29.
36. Ibid., 39.
37. Ibid., 59.
38. Ibid., 37.
39. Ibid., 59–60.
40. Benkler, "Coase's Penguin," 404.
41. Ibid., 412.
42. Tiziana Terranova, "Free Labor: Producing Culture for the Digital Economy," *Social Text* 18, no. 2 (2000): 33–58, quotation on 36. Terranova is citing Paul Gilroy, *The Black Atlantic: Modernity and Double Consciousness* (London: Verso, 1993), 40.
43. Maurizio Lazzarato, "Immaterial Labor," in *Radical Thought in Italy: A Potential Politics,* ed. Michael Hardt and Paolo Virno, Theory out of Bounds series (Minneapolis: University of Minnesota Press, 1996), 133–47; cited in Terranova, "Free Labor," 41.
44. Lazzarato, "Immaterial Labor," 133.
45. Ibid.
46. Ibid., 137.
47. Berry, *Copy, Rip, Burn,* 101.
48. Lazzarato, "Immaterial Labor," 137, my emphasis; and Terranova, "Free Labor," 42, my emphasis.
49. Benkler, *Wealth of Networks,* 16.
50. For case studies of the Apache and Mozilla projects, see Audris Mockus, Roy T. Fielding, and James D. Herbsleb, "Two Case Studies of Open Source Software Development: Apache and Mozilla," *ACM Transactions on Software Engineering and Methodology (TOSEM)* 11, no. 3 (2002): 309–46.
51. Christopher Kelty, "Geeks and Recursive Politics," in *Two Bits: The Cultural Significance of Free Software* (Durham, NC: Duke University Press, 2008), 27–63.
52. Ibid., 29.
53. Ibid.
54. See, for example, Gabriella Coleman's recent work on Hacker conferences ("The Hacker Conference: A Ritual Condensation and Celebration of a Lifeworld," *Anthropological Quarterly* 83, no. 1 [2010]: 47–72).
55. Weber, *Success of Open Source,* 171–72.

56. Galloway and Thacker, *Exploit,* 54.

57. An example of this can be seen in Charles Leadbeater's recent book *We-think:*

> The pioneers of open-source programming and online communities in the 1960s and 1970s talked the language of fellowship and communion. The web has never shaken off these communal and collaborative roots, which is why money-making companies have found it hard to bend it to their commercial purposes. (*We-think: Mass Innovation, Not Mass Production,* 2nd ed. [London: Profile Books, 2009], 48)

58. Alberto Toscano, "La disparation: Politique et sujet chez Simondon," *Multitudes* 18 (2004): 73–82. See also Alberto Toscano, "The Disparate: Ontology and Politics in Simondon," paper presented at the Society of European Philosophy/Forum for European Philosophy Annual Conference, University of Sussex, 2007, www.after1968.org/app/webroot/uploads/Toscano_Ontology_Politics_Simondon.pdf.

59. See Paolo Virno, *A Grammar of the Multitude: For an Analysis of Contemporary Forms of Life,* trans. Isabella Bertoletti, James Cascaito, and Andrea Casson, Foreign Agents series (Los Angeles: Semiotext[e], 2003).

60. From Muriel Combes, *Simondon individu et collectivité: Pour une philosophie du transindividuel* (Paris: PUF, 1999); quoted in Toscano, "La disparation."

61. Toscano, "La disparation."

62. Contrast Alberto Toscano's interpretation of the preindividual as disparation with Paolo Virno on Gilbert Simondon: "When we speak of a process, or a principle, of individuation, we should keep in mind what precedes individuation itself. This has to do, first of all, with a *pre-individual reality,* that is to say, something common, universal and undifferentiated" (Virno, *Grammar of the Multitude,* 76). Toscano's reading here, via Deleuze and Simondon, of *disparation* might be more productively compared with Geoffrey Bennington's understanding, via Hegel and Derrida, of *scatter* ("Scatter," *Oxford Literary Review* 30, no. 1 [2008]: 1–44).

63. Gilbert Simondon, *L' individuation psychique et collective: À la lumière des notions de forme, information, potentiel et métastabilité* (Paris: Aubier, 2007).

64. As Andrew Ross argues,

> [L]egally-minded coverage of IP [intellectual property] disputes tends to revolve exclusively around the interests of claimants: creators, copyright holders, or the general public of users and consumers. . . . Legal analysts of landmark cases rarely have anything to say about the multitude of jobs and livelihoods affected by the judicial treatment of IP-based assets and new technologies. Not only does this offend our sense of cultural and social history; it also weakens our capacity to understand, and react to, the vast changes occurring today as a result of the technology-driven IP property grab that has resulted in an aggressive expansion of copyright, patent, trademark, or publicity rights. ("Technology and Below-the-Line Labor in the Copyfight over Intellectual Property," *American Quarterly* 58, no. 3 [2006]: 743–66, quotation on 745)

I am indebted to the editors of this issue for pointing out to me this connection with Ross's work.